CLAIMS

1. A method of for automating the design of programmable logic devices, the method comprising the steps of:

obtaining design implementation files, and

calculating a set of design output files from the design implementation files without substantial intervention from a human operator.

2. The method of claim 1, wherein the design implementation files comprise a hollow netlist, filled netlist, and set of constraints.

3. The method of claim 1, wherein the design implementation files are not capable of programming the programmable logic device, and wherein the design output files are capable of programming the programmable logic device.

4. The method of claim 1, wherein the step of calculating a set of design output files comprises generating a set of scripts, setup files, and tool lineup files for use in programming the programmable logic devices.

5. The method of claim 1, wherein the step of calculating a set of design output files comprises the steps of initially placing logic groups on the programmable logic device, estimating the size of the logic groups, estimating timing for the placed and sized logic groups, and filling the logic groups with primitive information.

6. The method of claim 5, wherein the design implementation files comprise a hollow netlist and a set of constraints, and wherein the step of initially placing logic groups comprises merging the hollow netlist with the constraints, and iterating until the placement of the logic groups meets the constraints.

7. The method of claim 6, wherein a result of initially placing logic groups comprises a list of area groups.

8. The method of claim 5, wherein the design implementation files comprise a filled netlist and a set of constraints, and wherein the step of estimating the size of the logic groups comprises merging the filled netlist and constraints and iterating until the size of the logic groups is resolved.

9. The method of claim 8, further comprising the steps of analyzing the area usage of the programmable logic device and choosing an appropriate programmable logic device based on the usage analysis.

10. The method of claim 7, wherein the step of estimating timing for the placed and sized logic groups comprises merging the hollow netlist with the data-path constraints, and performing a timing analysis on the merged hollow netlist and data-path constraints to obtain an acceptable timing margin.

11. The method of claim 7, wherein the step of filling the logic groups with FPGA basic elements comprises merging the filled netlists and area groups, and running the filled design to verify that it will meet design criteria.

12. The method of claim 1, wherein the programmable logic device is a Field Programmable Gate Array.

13. The method of claim 12, wherein the design implementation files are files generated from Hardware Descriptor Language (HDL) files that have been subject to Register Transfer Language (RTL) Synthesis.

14. A computer configured to automate the design of programmable logic devices, comprising:

FPGA design software configured to translate at a file created using at least one of a Hardware Descriptor Language and a Register Transfer Language, into a format usable to program a programmable logic device.

15. The computer of claim 14, wherein the FPGA design software is further configured to iterate the translation to achieve an optimized format.